Developing the Systems Engineering Experience Accelerator (SEEA) Prototype and Roadmap

Final Technical Report

Principal Investigator: Dr. Jon Wade, Stevens Institute of Technology

Co-Principal Investigator: Dr. William Watson, Purdue University

Team Members

Dr. Douglas Bodner, Senior Researcher, Georgia Tech

Dr. Masa Okutsu, Purdue University

Dr. Alice Squires, Senior Researcher, Stevens Institute of Technology

Dr. Peter Dominick, Stevens Institute of Technology

Dr. Richard Reilly, Stevens Institute of Technology

Dan Ingold, Senior Researcher, University of Southern California

Research Assistants

With Contributions by Subject Matter Experts

Eric (Rick) Abell, John Griffin, and John McKeown

Contract Number: H98230-08-D-0171 DO1, TTO2, RT16

Report No. SERC-2011-TR-19

May 31, 2011

Report Documentation Page

Form Approved
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **31 MAY 2011**	2. REPORT TYPE	3. DATES COVERED **00-00-2011 to 00-00-2011**

4. TITLE AND SUBTITLE **Developing the Systems Engineering Experience Accelerator (SEEA) Prototype and Roadmap**	5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Systems Engineering Research Center,Stevens Institute of Technology,Castle Point on Hudson,Hoboken,NJ,07030**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR’S ACRONYM(S) 11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**31**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

This page intentionally left blank

EXECUTIVE SUMMARY

This document is a summary of the work that was completed in the first year of the SERC Research Topic DO1/TTO2/0016 “Developing Systems Engineering Experience Accelerator (SEEA) Prototype and Roadmap” supported by the Defense Acquisition University. The purpose of the research project is to test the feasibility of a simulated approach for accelerating systems engineering competency development in the learner. The SEEA research project hypothesis is:

> *By using technology we can create a simulation that will put the learner in an experiential, emotional state and effectively compress time and greatly accelerate the learning of a systems engineer faster than would occur naturally on the job.*

The major research activities that were completed in the baseline year are as follows:

1. Project Goals & Success Metrics Defined
2. Critical Competencies and Maturation Points Identified
3. Appropriate Learning Experiences Created
4. Open Architecture Defined
5. Technologies Selected
6. Prototype Developed
7. Prototype Demonstrated
8. Final Report Written (this document)

In addition to the work activities, four top program risks were identified and tracked throughout the first year of the program:

1. **Risk:** Inability to support known and evolving customer requirements with current staff, budget and timeframe. **Mitigation:** Build a detailed requirements list with effort estimations, and periodically review and re-prioritize the list with stakeholders identifying and resolving potential conflicts as they arise.

2. **Risk:** Inability to tradeoff the ability to rapidly create a prototype vs. a long term architecture and technology. **Mitigation:** Identify upfront the areas where the long term architecture and technology is unknown, or where it may be difficult to implement, and determine how and when the prototype implementation decision will be made and monitor throughout the prototype development process.

3. **Risk:** Inability to produce a prototype that provides a compelling experience, supports the desired learning and is seen to be authentic. This includes the development of dialogue and feedback to the Learner that are reasonable and plausible from both a behavioral and technical perspective. **Mitigation:** Develop

a success criteria trade-off framework and identify measures to track these success criteria during the development phase. Iteratively develop dialogue and feedback used during simulation that is based on inputs from SMEs. Have subject matter experts (SE and UAV) and representatives of the target learners go through the Experience throughout the development process providing continuous input.

4. **Risk:** Inability to successfully integrate our many ideas, approaches, requirements and developed technology and design. **Mitigation:** Employ a modular, loosely-coupled architecture that enables geographically-distributed developers to work independently.

At the end of the first year, a set of lessons learned were compiled and categorized into the following five areas:

1. Competencies, Learning and Content
2. Complexity/Effort vs. Authenticity/Learning
3. Technology
4. R&D Processes
5. Sponsor Involvement

Follow-on work has been defined for the next year in three phases:

- Phase 1: Update Documentation and Planning (June 2011)
- Phase 2: Prototype Development (July 2011 – November 2011)
- Phase 3: Prototype Validation (December 2011 – May 2012)

Subsequent work will involve measuring and analyzing pilot results.

TABLE OF CONTENTS

LIST OF FIGURES

PREFACE

This document is a summary of the work that was completed in the first year of the SERC Research Topic DO1/TTO2/0016 "Developing Systems Engineering Experience Accelerator (SEEA) Prototype and Roadmap" supported by the Defense Acquisition University. This summary focuses on each of the work items noted in the proposal.

The following are the documents that were produced by this research and may be referenced in this document:

- Experience Accelerator RT16 Project documents:
  - RT16 Project Goals and Success Metrics (A013)
  - RT16 Technical and Management Work Plan (A009)
  - RT16 Monthly Status Reports (A008)
  - Experience Accelerator Concept of Operations (A013)
  - Experience Accelerator System Architecture and Design Specification (A013)
  - Experience Accelerator Systems Specification (A013)
  - Experience Acceleration Experience Design Document
  - Developing Systems Engineering Experience Accelerator (SEEA) Prototype and Roadmap, Option Year 1 (Phase 1, Phase 2 and Phase 3)

- Publications:
  - Squires, A., Wade, J., Watson, B., Bodner, D., Okutsu, M., Ingold, D., Reilly, R., Dominick, P., Gelosh, D. (2011), "Investigating an Innovative Approach for Developing Systems Engineering Curriculum: The Systems Engineering Experience Accelerator", Proceedings of the 2011 American Society for Engineering Education (ASEE) Annual Conference and Exposition, Vancouver, BC, Canada, June 26-29, 2011.
  - Squires, A., Wade, J., Dominick, P., Gelosh, D. (2011) "Building a Competency Taxonomy to Guide Experience Acceleration of Lead Program Systems Engineers", Proceedings from the Ninth Annual Conference on Systems Engineering Research (CSER), Redondo Beach, CA, April 14-16, 2011.

1. INTRODUCTION

Systems engineering educators are struggling to address workforce development needs required to meet the emerging challenges posed by increasing systems complexity (Bagg, et. al, 2003) and the widening gap in systems engineering expertise in the workforce (Charette, 2008). The Systems Engineering Experience Accelerator (SEEA) research project was conceived as a critical response to these needs and challenges. The project was initiated to validate the use of technology to potentially create an experiential, emotional state in the learner coupled with reflective learning so that time is effectively compressed and the learning process of a systems engineer (SE) is significantly accelerated as compared to the rate at which learning would occur naturally on the job. The purpose of the research project is to test the feasibility of a simulated approach for accelerating systems engineering competency development in the learner. An example of how the various concepts developed for the SEEA are related is shown in Figure 1.

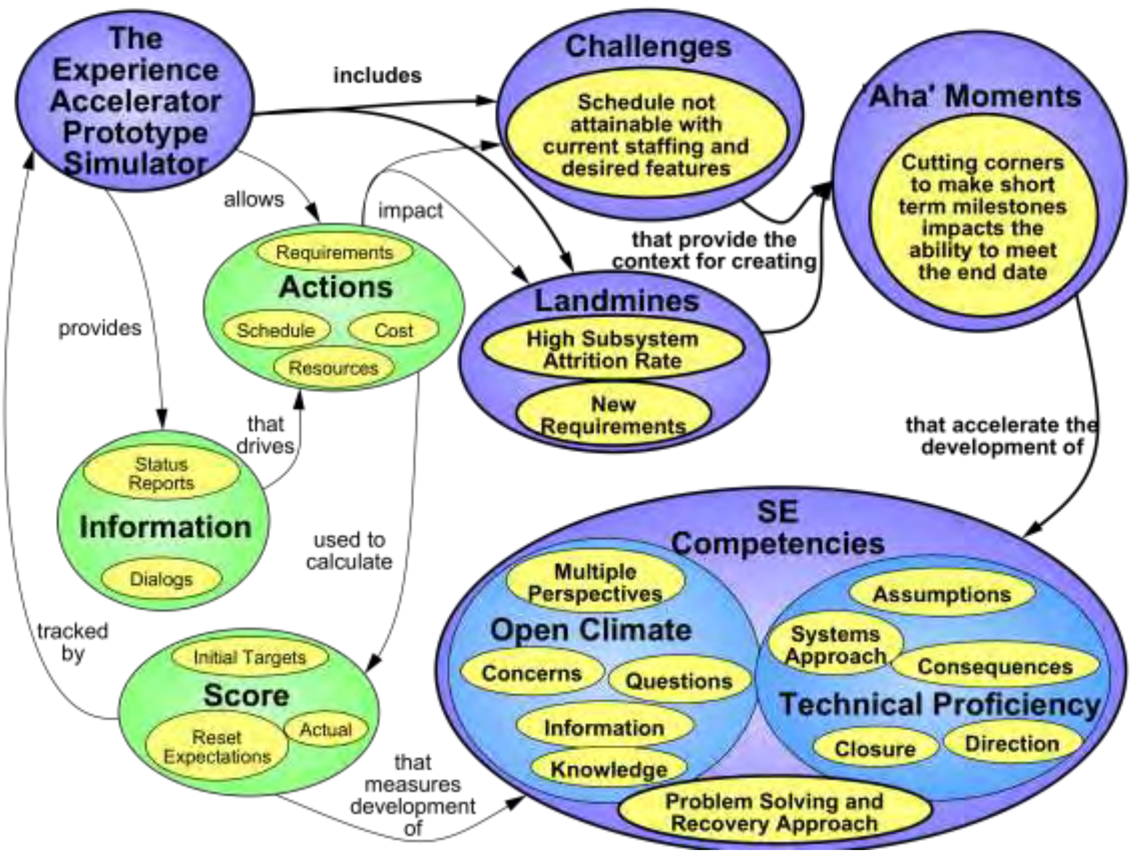

Figure 1: Notional Diagram of the SEEA Prototype Simulator

As shown, the development team had a threefold challenge to balance the development of the simulator technology that supports displayed content (shown in green) that, in turn, supports the developed concepts (shown in purple). The goal was to effectively create challenges and landmines that support the learner's experience of the necessary "Aha" moment. The intent was that by experiencing the "Aha" moment, the learner transitions to a more advanced level of understanding in the targeted competency, in this case "Problem Solving and Recovery Approach".

The testing of the prototype will support evaluation of the theoretical capabilities of the developed system and provide guidance for the continuing development of the SEEA simulator going forward. The initial learners of the simulator will be members of the acquisition workforce in training at the Defense Acquisition University (DAU). The simulator is also planned as a component of graduate level technical leadership programs and executive level training.

1.1 Background and Motivation

In *The Art and Science of Systems Engineering*, Mr. Harold Bell, Director Advanced Planning and Analysis Division, NASA Office of Chief Engineer, is quoted as saying: "A great systems engineer completely understands and applies the art of leadership and has the experience and scar tissue from trying to earn the badge of leader from his or her team" (Ryschkewitsch, et. al, 2009, p. 2). Historically, competent systems engineers have developed their "scar tissue" by gaining the necessary insights and wisdom through both failures and successes, in an integrated real world environment. In the workplace, however, the learning events that result in the development of "scar tissue" are distributed, sometimes sparsely, over time. In addition, a common benchmark time for the development of a competent SE is a minimum of about 10-15 years (Dubey, 2006). Given there is a shortfall of SEs in the global workforce today (NDIA SE Division, 2010) and no readily available source of SEs to replace the top SEs in the retiring baby boomer generation, the time to develop competent SEs needs to be significantly shortened.

The primary goal of the SEEA, once it is developed, is to accelerate the maturation of SEs in the workforce by providing the opportunities to earn "scar tissue" through realistic, engaging simulation. These tailored experiences will allow the learner to feel the consequences of success and failure in a simulated environment so they can gain the necessary insights and wisdoms to mature as a SE, and yet not jeopardize the lives of others or compromise their careers. The initial target audience of the SEEA program is lead program SEs in the acquisition workforce who are required to effectively manage complex systems throughout their lifecycle from an acquisition/acquirer viewpoint in a typical program office. The initial focus is on maturing these leads to prepare them for executive assignments.

1.2 Project Goals and Success Metrics

Based on SEEA research team meetings and feedback provided by the sponsors, the team set specific goals and success metrics as summarized in the following sections.

1.2.1 Purpose

The ultimate purpose of the SEEA is to leverage technology to create an experiential, emotional state in the learner so that time is effectively compressed and the learning process of a systems engineer accelerated as compared to the rate at which learning would occur naturally on the job.

The purpose of this project is to develop a prototype of the SEEA that is focused on a small set of competencies, in order to evaluate the theoretical capabilities of that technology.

1.2.2 Program Goals

The primary goal of the SEEA is to transform the development of systems engineers by creating a new paradigm capable of significantly reducing the time to mature and sustain a senior systems engineer while providing the skills necessary to address emerging systems challenges in an economically attractive manner. Outcomes needed to achieve this goal include:

1. Moving the systems engineer to the next level of proficiency in one or more SE competencies as listed in the Systems Planning, Research Development, and Engineering (SPRDE) Systems Engineering (SE) and Program Systems Engineer (PSE) competency model, known as the SPRDE-SE/PSE.
2. Developing and maturing systems thinking skills.
3. Developing and maturing leadership skills.

1.2.3 Target Audience

The initial focus is on the Systems Engineering Executive Level skills of a DoD Lead Program Systems Engineer necessary to effectively manage complex systems throughout their lifecycle from an acquisition/acquirer viewpoint in a typical Project Management Office (PMO). The skills addressed may well complement or support those taught in senior program management courses. The SEEA targets the entire life long learning of the Systems Engineer.

1.2.4 Success Metrics

Success of the year one prototype will be indicated with a positive result in the following areas:

- Experienced Lead Program Systems Engineers authenticate the SEEA and provide useful feedback on areas of improvement.
- Learners have identified that it has a significant favorable job impact.
- The potential for learners that successfully complete the training to be able to immediately implement lessons learned from the training experience to the job, assuming the culture allows this.
- The potential for PSEs to be able to perform targeted Level 3 competencies at one or more higher levels of proficiency.

1.3 TECHNICAL RISK/MANAGEMENT PLAN OVERVIEW

The RT16 work plan is summarized in the next section, with the detail in the *Technical and Management Work Plan (A009)*. Program risks, addressed in the following section, were also reported on in the latter half of the project in the Monthly Status Reports (A008).

1.3.1 WORK PLAN

The major research activities that were completed in the baseline year are as follows:

1. **Project Goals & Success Metrics Defined:** The project charter addressed four areas: 1) Purpose; 2) Goals; 3) Target Audience; and 4) Success Metrics. These are summarized in the previous section on *Project Goals and Success Metrics* and defined in the *RT16 Project Goals and Success Metrics (A013)* document.

2. **Critical Competencies and Maturation Points Identified**: This activity involved the identification of critical knowledge and competencies for maturing systems engineers including typical human failings in multiple domains of systems engineering that could potentially lead to program failure. Primary research was performed through a series of interviews with systems engineering subject matter experts. Systems engineering experts were asked about 'Aha' moments or turning points in their career where they developed insights that helped them mature as systems engineers. Data collected included low and high points of their career that led to important insights and lessons learned related to their maturity as systems engineers, and the context within which the 'Aha' moments occurred. Secondary research included an extensive review of published literature on the topic. In support of this activity, the SEEA team completed the following components of the design, each detailed further in the section on *Research Approach*:

    - An integration of three competency models into a competency taxonomy (see *Competency Taxonomy*);

- An approach for creating a learner profile that could be used to tailor the experience (see *Learning Theory Model and Learner Profile*).
- A collection of 'Aha' moments based on a series of interviews with systems engineering subject matter experts that included the context within which the 'Aha' moments occurred and a set of simulated challenges and land mines for the learner to experience (see *'Aha' Moments, Challenges, and Landmines*).

These activities are also described in more detail in the references: Squires, et. al., 2011; and Squires, Wade, Dominick, and Gelosh, 2011.

3. **Appropriate Learning Experiences Created:** This activity involved the creation of learning experiences and associated objectives that could most effectively reinforce the desired lessons and increase the competency and maturity level of the participants. It is critical that these experiences are sufficiently interesting, have the necessary fidelity and provide sufficient intellectual and emotional content so that they result in lessons learned, so that the practitioner can carry out appropriate actions later in time. This is a highly creative aspect of this program which will continue throughout the program. As the experience narrative was developed, its effectiveness has been assessed by experienced subject matter experts. A learning process was also defined to illustrate how learners will learn through these experiences (see *Learning Theory Model and Learner Profile*). The specific learning experience created for the SEEA prototype are detailed in the *Concept of Operations (CONOPS)* and the *Experience Design* documents and summarized in the corresponding sections of the Final Report.

4. **Open Architecture Defined:** The creation of a Concept of Operations (CONOPS) for the Experience Accelerator set the requirements for this work. The plan was to use an open architecture to provide the ability to rapidly and inexpensively develop and iteratively improve and expand learning materials. The architecture was developed seeking to allow for the independent evolution of the constituent technologies, enabling long-term support and feature enhancement. That is, the intent was not to preclude the development of an open source vehicle for distributed innovation. The chosen open architecture has been documented appropriately as a deliverable of this work and has been updated periodically throughout the research process as it has evolved. While the Year 1 Prototype did not identify an open architecture as being essential for the first year proof of concept prototype, every effort was made to utilize an open architecture while not jeopardizing the completion of the project. The ultimate architecture of the SEEA prototype will be further refined to be as open as possible. The modularization of the prototype was defined in order to specify how the prototype will function and how its different features interface. The

System Specification and *System Architecture and Design* documents provide the detail on the open architecture and technologies selected, and these documents are summarized in the corresponding sections of the Final Report.

5. **Technologies Selected**: A review of potential technologies were reviewed and evaluated according to a number of factors, including: ease of development, cost, availability of support and established development community, match with functional requirements, and adherence to open source architecture. A further review was conducted to evaluate other technologies currently being used on DAU sponsored projects, specifically the Nexus Virtual World project. As this project has been active for a number of years, the technologies it utilizes have already been cleared for use on client computers. Furthermore, the architecture chosen for this project focuses on delivering web-based experiences on nearly any computer, regardless of hardware. These considerations, along with the desire for future collaboration between the two projects led to the selection of similar technologies for the SEEA project. The SEEA prototype was developed primarily utilizing Adobe Flash, XML, and Java. While it is not intended that the *Systems Specification* is fully complete, it establishes a first draft at a functional development baseline that will be updated after feedback has been received based on the *Prototype Demonstration*, and thus will form the basis for the follow-on prototype evolution.

6. **Prototype Developed:** The two final activities were the prototype development and prototype demonstration (see 7. next). The prototype development work addressed two major areas: technological development and content development. First, technologies had to be identified, selected, and integrated into the defined open architecture to provide the necessary infrastructure. From this as a basis, missing components were defined, developed, tested and validated to ensure the necessary level of fidelity. For example, the simulation engine was constructed using the Abdel-Hamid model as a basis. Capabilities were added to the model in a number of important areas including the ability to support subprojects over an entire lifecycle, to calculate cost and model completion of project capabilities such as Unmanned Aerial Vehicle (UAV) range. In addition, a virtual 2D desktop was created to simulate the PSE's desktop. This includes a text editor, calendar, email and phone system. The second major area of work was the development of particular content for the prototype demonstration. This is a production step that defined the content needed to complete the learning experience. The content was incrementally evaluated both by the sponsors as well as by a set of subject matter experts. The plan is for future content and experiences to be evaluated from members of the research and sponsor communities. This completed work should be usable for a variety of learning experiences. While the infrastructure may not be complete,

the critical areas of work were identified and supported with a prototype implementation.

7. **Prototype Demonstrated:** The demonstration of the prototype to the sponsors addressed the feasibility of the architecture, technology and content approach in the selected areas of learning. Ultimately, it was decided that the Year 1 prototype would consist of a 7-phase experience (not including an initial orientation), where the Learner assumes the role of a PSE in a UAV acquisition project. The Experience covers the project lifecycle from post Preliminary Design Review (PDR), through system development and demonstration (SDD), through integration and field testing so that the Learner can see the impact of his/her decisions. The total length of the prototype Experience is estimated to be approximately 2.5 hours (better estimated will be made after trials with trial learners); however, the Learner can go through the Experience more than once. (This duration is limited to facilitate prototype development and test, and is in no way intended to be a limitation for a deployable Experience.)

8. **Final Report Written**: This document represents the final report.

1.3.2 Risk Management

In addition to the work activities, four top program risks were identified and tracked throughout the first year of the program in the following areas:

1. Project Management
2. Technology Development
3. Content Development
4. Integration

Mitigation strategies were put in place as outlined in detail in the following sections.

1.3.2.1 Risk 1: Project Management

Risk 1: Inability to support known and evolving customer requirements with current staff, budget and timeframe.
Mitigation: Build a detailed requirements list with effort estimations, and periodically review and re-prioritize the (possibly evolving) list with stakeholders identifying and resolving potential conflicts as they arise. Incrementally implement and continuously integrate requirements in priority order, to ensure that final product incorporates the most important requirements that can be implemented with the available resources, in the available time. As necessary find additional resources to do development work to fill any identified gaps. In addition, extend the development time out until the May to June timeframe with a zero cost program extension.

1.3.2.2 Risk 2: Technology Development

Risk 2: Inability to tradeoff the ability to rapidly create a prototype vs. a long term architecture and technology.
Mitigation: Identify upfront the areas where the long term architecture and technology is unknown, or where it may be difficult to implement, and determine how and when the prototype implementation decision will be made. Monitor this throughout the prototype development process. Document the long term architectural needs and approaches vs. the short term prototype needs so that these can be reviewed and explicitly traded-off.

1.3.2.3 Risk 3: Content Development

Risk 3: Inability to produce a prototype that provides a compelling experience, supports the desired learning and is seen to be authentic. This includes the development of dialogue and feedback to the Learner that are reasonable and plausible from both a behavioral and technical perspective.
Mitigation: Develop a success criteria trade-off framework and identify measures to track these success criteria during the development phase. Iteratively develop dialogue and feedback used during simulation that is based on inputs from SMEs. Have subject matter experts (SE and UAV) and representatives of the target Learners go through the Experience throughout the development process providing continuous input. Leverage the team's behavioral learning expertise during this process. Produce dialog and artifact authoring tools which allow these activities to take place independently of the technology development process.

1.3.2.4 Risk 4: Integration

Risk 4: Inability to successfully integrate our many ideas, approaches, requirements and developed technology and design.
Mitigation: Employ a modular, loosely-coupled architecture that enables geographically-distributed developers to work independently. Define explicit, arms-length interfaces between modules to simplify integration. In particular, create interfaces and tools (as noted above) that allow the development of content to proceed independently of the technology development. Use continuous integration as part of an agile development process to identify and resolve integration problems early ("fail fast").

2. RESEARCH APPROACH

2.1 HYPOTHESIS

The SEEA research project hypothesis is:

> *By using technology we can create a simulation that will put the learner in an experiential, emotional state and effectively compress time and greatly accelerate the learning of a systems engineer faster than would occur naturally on the job.*

2.2 COMPETENCY TAXONOMY

The SEEA research team chose to combine the following three models into a single competency taxonomy for the project:

1. The Systems Planning, Research Development, and Engineering (SPRDE) Systems Engineering (SE) and Program Systems Engineer (PSE) competency model, known as the SPRDE-SE/PSE.

2. The SERC Technical Lead Competency Model (Gavito, et. al, 2010)

3. A Critical/Systems Thinking Competency Model (Squires, 2007)

The final SEEA competency taxonomy has six primary groupings, as shown in Figure 2, that are further divided into two to six competency areas that contain a total of 87 unique competencies.

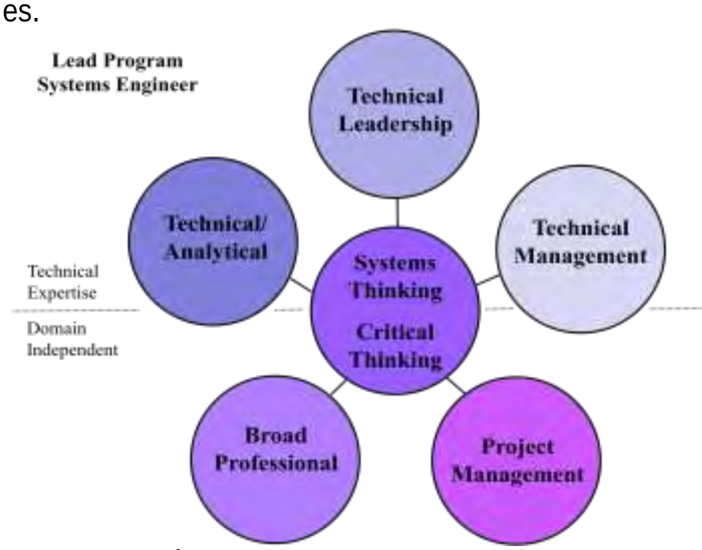

Figure 2: SE Competency Taxonomy

The SEEA prototype focused on the 'Problem Solving and Recovery Approach' competency within the 'Broad Professional' area (see Squires, Wade, Dominick, and Gelosh, 2011 for more detail). The model includes a proficiency table that measures the learner's proficiency level in each competency based on the complexity of the system being simulated and the learner's level of demonstrated ability to apply the competency for each level of complexity.

2.3 Learning Theory Model and Learner Profile

The SEEA was conceived to help learners learn from mistakes in an environment in which they can face challenges and make errors without there being any long term negative work-related outcomes. However, learners must receive clear and actionable feedback as the basis for reflection, subsequent skill practice and behavior change. Therefore, performance assessment and feedback are important aspects of the overall learning process. As depicted in Figure 3, that process can be described in relation to the Experiential Learning Model developed by Kolb, 1984.

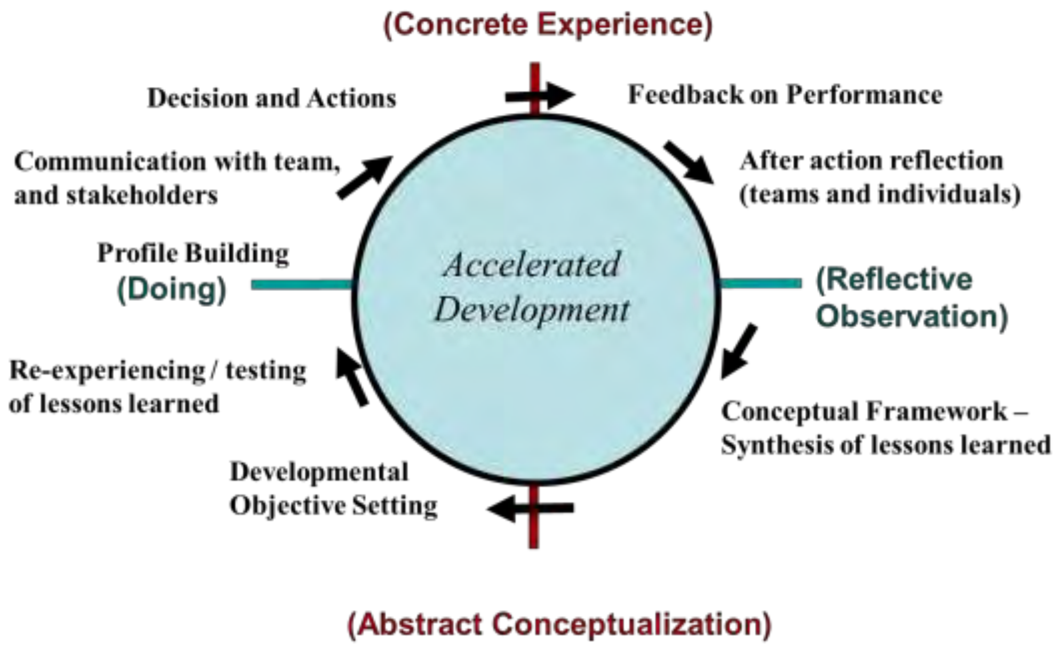

Figure 3: Learning Process: All Phases of Experiential Learning to be Engaged

Profile building engages learners in initial reflection on their related skills, personal qualities and experiences. Profile building is followed by the *concrete experience* aspects of the simulation such as communication and interaction with other players, non-player characters, and scenario content as well as actual learner decision making and actions. Those decisions and actions then become the basis for assessment and feedback, thereby moving the learner into the next phase of the experiential learning process, r*eflective observation*. Feedback and reflection then becomes the springboard for helping individuals engage in *abstract conceptualization*. In support of this framework, prior to beginning their work with the SEEA, learners are asked to provide information about their personal qualities that are likely to influence how they would approach the scenarios and challenges the accelerator will pose. For year one, this assessment focused on collecting learner feedback on their confidence level in attributes that support the 'Problem Solving and Recovery Approach' competency.

2.4 "AHA" MOMENTS, CHALLENGES, LANDMINES

'Aha' moments (and associated context) were combined with collected learner profile information that included the learner's perception of their own competency levels, to drive the types and difficulty levels of challenges and land mines in the simulation. The Office of the Deputy Under Secretary of Defense for Acquisition, Technology & Logistics (AT&L) uses the Program Support Review (PSR) process to assess the program systems engineering (PSE) policies and practices of programs under its authority. A PSR assessment is performed according to the Defense Acquisition Program Support (DAPS) Methodology (Defense Acquisition Program Support (DAPS) Methodology, 2009), designed in 2004 and revised in 2009, which is used to organize the review process. The results of many years of PSR assessments using this methodology have in turn been compiled into a database of the systemic root causes of program issues.

The DAPS Methodology is based on the policies set forth in the Defense Acquisition Guidebook, and is focused on Guidebook, and is focused on preparing program management for milestone decision reviews, particularly with reviews, particularly with respect to the systems engineering aspects of the procurement process. Assessments process. Assessments are organized into six areas: mission capabilities/ requirements generation, resources, generation, resources, management, technical process, and program performance. These areas are in turn These areas are in turn divided into sub-areas listing the principal factors contributing to each area, and to each area, and providing evaluation criteria and question sets to facilitate in-depth discussion. The Experience discussion. The Experience Accelerator scenarios should reflect the types and range of issues that real-world issues that real-world programs face. Given the comprehensive approach of the DAPS Methodology, its close tie-Methodology, its close tie-in to defense acquisition policy, and its focus on systems engineering, it appears to be a engineering, it appears to be a reasonable yardstick against which to measure the breadth of challenges offered by breadth of challenges offered by the Experience Accelerator.

Figure 4 compares the 68 challenges, landmines, evidence, and actions in the Experience Accelerator against the areas of the DAPS Methodology.

It can be seen that the present SEEA scenario concentrates on resource and management areas, focuses less so on mission capabilities and technical process areas, and omits references to the area of performance. While not ideal, this mix reflects the nature of the SEEA prototype, which chose to focus on constructs that can be easily represented in an interactive simulation context.

The resources and management areas predominate because these are areas where the systems engineering learner can easily express recommended changes within the SEEA, and where the effects of these changes can be practically modeled. The reallocation of resources—primarily staffing but also budget—reflects some of the simpler changes the learner might affect. Management changes, which include schedule modifications and requests for additional capital, are also common (though perhaps less desirable) changes that a systems engineer might recommend.

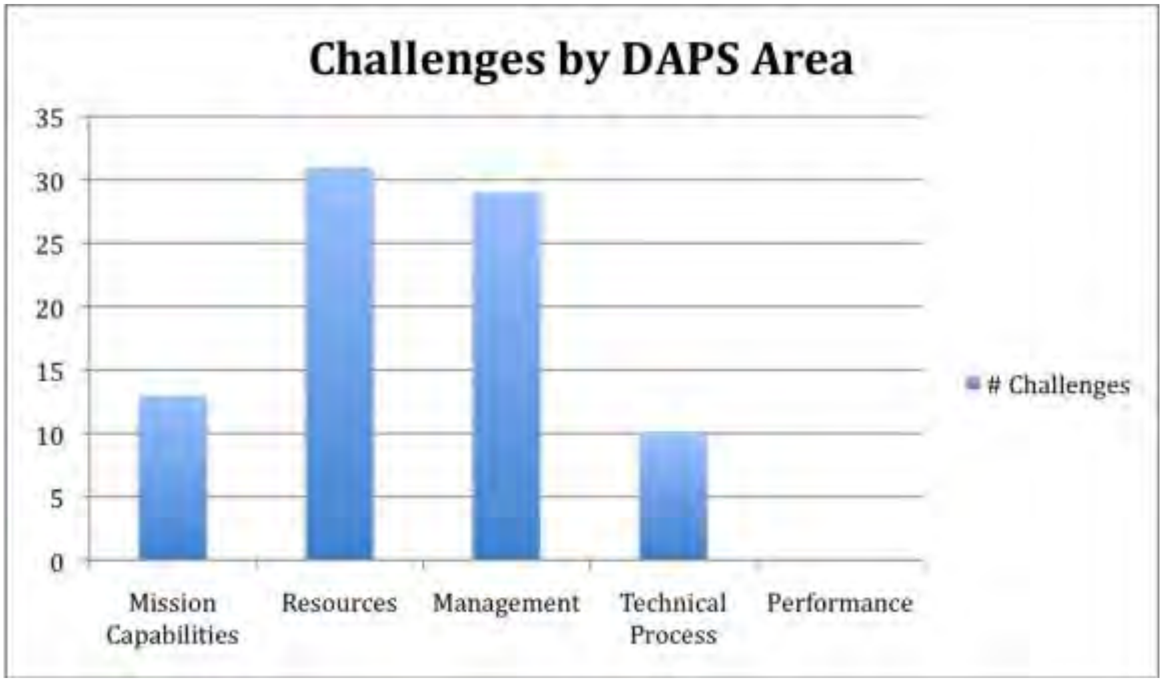

Figure 4: SEEA Challenges by DAPS Area

Reductions in mission capabilities might be required where technical hurdles are insurmountable, and are one of the mitigating actions the learner can recommend. It is difficult to represent in the prototype SEEA, however, the complex political choices and tradeoffs that occur when such reductions are contemplated. The SEEA simplifies the choice of mitigating effects to recommending the reduction of capabilities, and avoids analysis of the larger systemic and political issues.

Similarly, the technical process area of the DAPS methodology discusses how programs are technically executed—that is, the design of processes to ensure satisfactory program outcomes. It is not clear how to express this meta-process—the process one uses to decide the processes a project should employ—in the SEEA simulation. Instead, technical process challenges in the SEEA are primarily mitigated by resource reallocation—for example, by focusing resources on design reviews when defect rates are high—whereas a real-world solution might be to improve the design process.

The final area of the DAPS Methodology, performance, reviews the non-functional requirements of a project—its "-ilities"—suitability, survivability, effectiveness, and producibility. These issues again are addressed in the real-world by introducing the proper processes into a program, which ensure that non-functional requirements are properly implemented and verified. A future version of the SEEA might teach both performance and technical processes by allowing the learner to choose the types of processes to implement, based on the needs of the simulated program.

3. CONCEPT OF OPERATIONS

The SEEA represents a new means of maturing systems engineers in significantly reduced time from that normally required to reach a senior level of experience. The new experience-based training complements more traditional means of knowledge acquisition, and is designed to train and sustain the Systems Engineering workforce, while providing the skills necessary to address emerging systems challenges in an economically attractive manner. Anticipated outcomes include moving systems engineers to the next level of proficiency in one or more SE competencies and increasing their ability to apply these skills effectively on the job.

The Experience Accelerator is intended for lifelong learning of the Systems Engineer providing:

- A supplement to other types of education and training
- General job-related experience
- Specific contextualized job experience
- A measure of the compatibility of the learner to a specific role and responsibility at the current time; and a measure of the potential for growth into new roles and responsibilities moving forward

The following are some of the operational environment and constraints for the prototype version of the SEEA:

- **Personnel:** The Experience Accelerator will provide support for single-learner experiences through the use of artificial intelligence to support the non-player characters (NPCs), mentoring, profile generation support and experience training for learner.

- **SE Training Time:** The Experience Accelerator will provide the ability for the learner to have sessions that last no longer than one hour each that can be experienced from work, home or on the road. There will be no requirement for any other person to be online; however, the learner will need network access.

- **Technology:** The Experience Accelerator will not require any special hardware or graphics acceleration on the client hardware. The application will be web based to enable the use of light weight clients with slow network bandwidth. The Learner may partake in the Experience at work, on the road or at home.

- **Physical Plant:** No physical classroom or text books will be required.

- **Funding:** The Experience Accelerator will develop or use open source content as much as possible so that there will be little or no licensing needed. The goal is to create an open source foundation and community to provide a rich set of curriculum, content and technology that can be freely used and leveraged.

For more detail, please see the *Experience Accelerator Concept of Operations (A013)*.

4. System Architecture and Design

The major modules of the SEEA, as shown in Figure 5, are:

- **Experience Master:** contains the overall Experience state and provides control and sequencing for the other major EA modules.

- **Challenge Control:** contains the Learner profiles and Experience history logs and leverages these in conjunction with the competency taxonomy and 'Aha' moments to determine the appropriate challenges and landmines for each Learner

- **Simulation Engine:** determines the future state of the system and outputs to be presented to the Learner,

- **Non-Player Characters (NPC) Engine:** represents other non-player characters in the simulation, and creates and assembles the content for Learner interactions, and

- **Presentation Engine:** accepts inputs from the Learner and provides the presentation of the Experience interface to the Learner.

A single internal interface bus is provided to share information between these four major modules. Isolation layers are provided around the Presentation Engine to facilitate support of changing engine technologies. Finally, Experience generic technology that can be used to support multiple domains and experiences, and experience-specific blocks are segregated to support an open architecture with the intent to maximize reuse. Except for the simulation models, all of the Experience Specific information is stored in databases within the system.

All interactions with the Learner take place via the Client Isolation Layer API contained in the Presentation Engine module. The Simulation Engine determines the information that will be presented to the Learner, the NPC Engine formats and stores the information and processes the transactions, while the Presentation Engine creates the appropriate look and feel for the interaction. For more detail, please see the *Experience Accelerator System Architecture and Design Specification (A013)*.

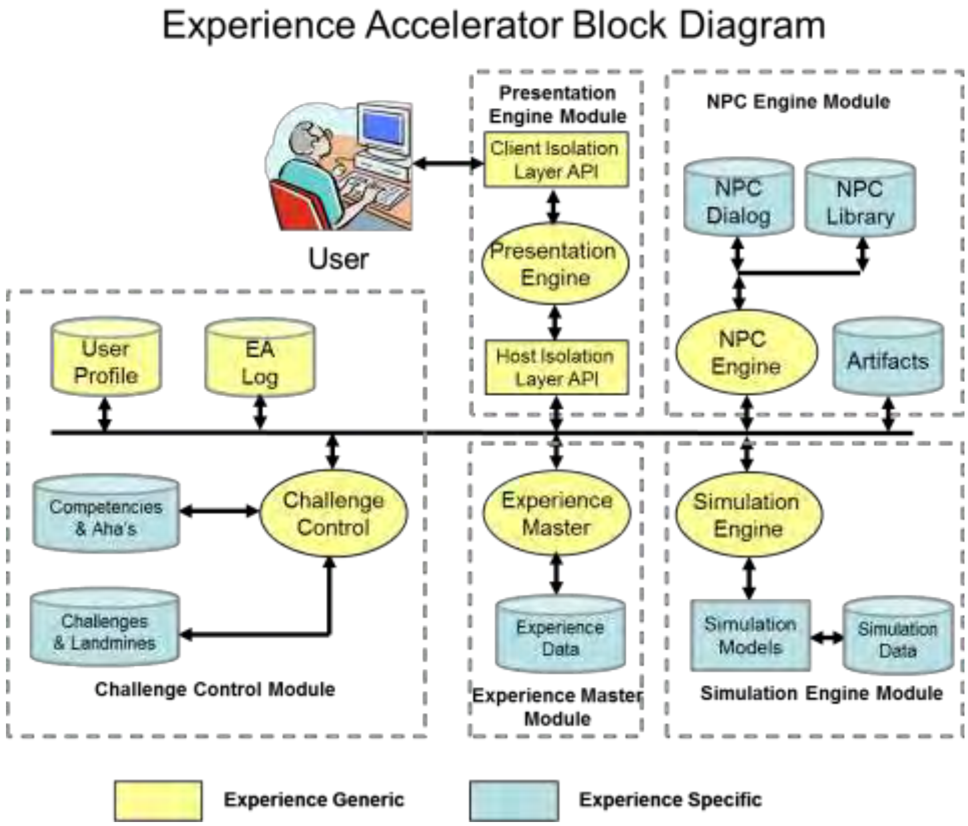

Figure 5: Experience Accelerator Logical Block Diagram

5. SYSTEM SPECIFICATION

The system specification documents the actual implementation of the system. For the prototype development of the SEEA, our team employed agile development methods, iteratively and incrementally developing the system based on frequent feedback from the team members and the sponsors. Because the prototype was developed by geographically distributed teams, a development server with version controls as well as online communications (e.g., teleconferences and emails) was vital in facilitating the integration of the contents and different components of the software with the presentation engine itself.

The architecture of the system can be divided into two main categories, which are the server and the client. The server keeps the information of the learner profiles, the content and handles heavier calculation; while the client parses and displays the information as well as acts as an input-output device between the learner and the system.

The server and the client are connected to each other through regulated means, which protects the information in the server so that it cannot be edited or accessed from the client in ways that are not designed. The aim of this architecture is to keep the client a light program that can operate via a web browser without requiring a high-end computer. Also, while our prototype assumes a 2D learner interface, one of the baseline assumptions was the ability to evolve into 3D solution, if so desired. For this reason, we ensured that our software can support 3D graphics. Compatibility to other projects (e.g. NEXUS) has been verified as well.

For more detail, please see the *Experience Accelerator Systems Specification (A013)*.

6. EXPERIENCE DESIGN

The experience design documents the initial experience and the content for that experience that was developed to demonstrate the prototype SEEA. The prototype operates in single-learner mode and each session lasts about 15 minutes with the first sessions in a phase lasting longer and later sessions taking less time. For the first year prototype, the total Experience target completion time is 2.5 hours or less to reduce the time necessary to validate the concept.

The SEEA prototype operates as follows. First the learner logs into the system and is presented with the control screen. Next, the learner selects either the UAV experience (developed for the prototype) or profile update, and continues. The UAV experience is made up of seven phases as follows:

- EA Introduction:
    - Phase 0: New Employee Orientation
- Experience Introduction:
    - Phase 1: New Assignment Orientation
- Experience Body:
    - Phase 2: Pre-Integration System Development
    - Phase 3: System Integration
    - Phase 4: System Field Test
    - Phase 5: Limited Production and Deployment
    - Phase 6: Experience End
- Experience Conclusion:

- Phase 7: Reflection

For more detail, please see the *Experience Acceleration Experience Design Document*.

7. Forward Plan

7.1 Lessons Learned

The following is a summary of the lessons learned from the RT16 team. The lessons are divided into the following five categories:

1. Competencies, Learning and Content
2. Complexity/Effort vs. Authenticity/Learning
3. Technology
4. R&D Processes
5. Sponsor Involvement

7.1.1 Competencies, Learning and Content

It was more difficult than expected to have competency models drive simulation design. While it is possible to create a clean taxonomy of competencies, these are actually blended in actual learning scenarios. We found that there was considerable overlap amongst some of the competencies such that they could not be clearly defined and addressed.

It is difficult to assess competency at the behavioral level in the simulation without having samples of actual learners who have gone through the experience. Ideally, we could define performance at the behavioral level with a small sample of learners who could be interviewed post simulation. This would allow both assessment and feedback to be more effectively linked to specific behaviors elicited by the simulation.

It was critical to have SMEs engaged to provide rapid feedback on the learning experiences and content. Time needs to be allocated to have the SMEs participate in the experience and provide feedback to ensure that the proper learning outcomes are reinforced.

Simulating interpersonal aspects of systems engineering was found to be very challenging, certainly more difficult than anticipated. This was particularly true due to the many possible situations that might arise due to the non-canned nature of the simulation (see Complexity vs. Authenticity below).

7.1.2 Technology

Open source technology provides numerous benefits in terms of allowing a learner community to form around a software product/system, but involves significant work to create a critical mass of functionality, especially when the software product/system integrates numerous individual technologies. It is fortunate that we did not create a dependence on NEXUS or other proprietary technologies.

The choice of software design suite and the programming languages is important, as it largely determines the nature and the scope of the development effort. There is a classic dilemma, however: the best tool set will be evident if the goals are clearly defined in detail, yet defining exactly what to develop will require some assumption about technologies. As in many development projects for training software, we went through a few iteration of this "chicken and egg" problem.

It is a good thing that we restricted ourselves to a 2D environment, as the issues with supporting a 3D space would certainly not be worth the effort at this phase in the R&D process.

Systems dynamics approaches can be used to model a variety of systems including those with smooth flows and those that are discrete (such as stochastic quality issues which can be supported through a look up table approach). We will need to explore when and how systems dynamics approaches are best applied and how they might be used with randomization and discrete event techniques.

Creating a professional look and feel for a virtual desktop is not a trivial undertaking. We will need to do some studies to determine what else we need to do to make it more intuitive and easy to use (along with providing the appropriate help documentation).

7.1.3 Complexity/Effort vs. Authenticity/Learning

Defense acquisition is a very complex enterprise, with many processes, actors and organizations. Selecting a subset of these to represent in the Experience Accelerator involves numerous design trade-offs to address the interests of (i) the learner community, that wants a realistic but not overwhelming experience, (ii) the education community, that wants realism in support of learning objectives, (iii) the acquisition community, that wants its various aspects represented faithfully, and (iv) the developer community, that wants to provide a useful product while managing complexity, schedule and cost.

One of the biggest challenges was to create an authentic, learning experience while managing complexity and the amount of content that needed to be created. We discovered the following eight major leverage/complexity points:

1. **Challenges & Landmines**: There are an almost infinite number of ways in which a program can fail; combinatorial explosion is a major challenge. This is not so much of a challenge for the simulator, but it is a major issue for the creation of artifacts and dialog which can support these to allow the Learner to make sense of the situation. While we created a catalog of a large number of frequently encountered Challenges and Landmines, for the prototype we implemented just a few of the most likely ones in the areas of aviation hardware and software.

2. **KPMs:** KPMs drive the amount of information that needs to be simulated, the amount of artifacts for background information, dialog and Learner recommendations. As such, we decided to limit these to schedule, quality and capabilities (range). Another important one is cost which will require the development of a plausible cost model and supporting artifacts and dialog. This development will be high on the priority list for work in Option Year 1. *Creating a consistent Cost model and integrating with the IMS/Schedule of events will be a major challenge and will require access to existing cost data for some DOD Programs.*

3. **Phases:** How many project phases are required? For this experience we omitted the effort necessary to create the program and focused on problem discovery and recovery. However, we kept the development, integration, field testing and production in place so that the Learner can see the downstream effects, but limited the detail in the later phases. Unfortunately, each of these phases has very different requirements for the development of simulation and content.

4. **Cycles per Phase:** The more cycles per phase, the greater the amount of Learner interaction and the need for additional dialog. If this Experience had been multi-learner, then this would have been less of a challenge, but with NPCs in all of the roles, this requires the creation of a good deal of additional dialog. More cycles per phase are not an issue for the other development areas such as simulation and Learner recommendations. It was felt that three months per cycle is optimal as longer cycles do not provide sufficient points for Learner interaction and shorter cycles do not provide enough information for action.

5. **Reviews:** There are a great number of reviews that are required in a large DoD acquisition program. We attempted to reduce the number and focus on only a restricted number of areas. To do otherwise would require the development of a great deal of background information on a project which might well be classified.

6. **Feedback to/from Learner**: In real life, there are many forms that this can take through documents, presentations, email, phone calls, tele/video conferencing and live meetings. These all require development efforts. To reduce this effort, we limited the communications to documents, email (receive only) and phone calls. We will need to determine if this is sufficient in Option Year 1.

7. **NPCs:** NPCs require a substantial amount of work with respect to creating dialog (see below). To simplify this we have three major NPCs – PM, PSE Prime contractor and the Mentor. There are a few other minor roles in the Ex-PSE (Phase 1 only), Government Test (Phase 3 only) and some EA administrators (only in Phase 0).

8. **Dialog:** Creating authentic and meaningful dialog is one of the largest challenges for this program given that all roles besides that of the Learner (PSE) are supported by Non-Player Characters (NPCs). The challenge is to minimize the total effort, or at least keep it tractable given the huge space of potential responses. We have decided to use a hub and spoke approach to avoid the need for speech recognition, yet hopefully provide an authentic set of responses. Within the dialog we are using "SelectIf" conditionals such that response selection is based on the simulation results. There are a number of other techniques that we can employ in the future to assist with this. One approach is to use variables based on simulation results for the NPC's responses such as, "We have fallen x.x months behind in the schedule." We also want to track the number of spokes that are traversed such that the conversation will 'time out' and prevent the Learner from exploring all of the questions. We also would like to be able to keep track of which spokes have been explored which can be used for feedback and also to allow the NPC to have a different response if the same spoke is traversed more than once. This is a broad space for future exploration and research.

7.1.4 R&D PROCESSES

At the outset of this project, given the exploratory nature of the research, open communication was established between all of the team members through the use of a Wiki site. However, it was discovered that the overhead in learning how to use and navigate through the site outweighed its advantages so we migrated to the use of a simple DropBox technology for documents and development code, and Webex for group meetings which were held once a week. As the team more than doubled in size and became more specialized, this mode of communication became a bit cumbersome. We eventually migrated to a weekly meeting with the SMEs for content, a weekly team meeting and technology meetings on an as needed basis.

Moving forward as the research has come much better defined, communication needs to be better tailored to the individual who needs the information at the time they need it rather than through information broadcast. This should be done through the formation of teams assigned to specific deliverables along with more decentralization of the efforts. The key team members will agree on a management and a communication plan for Option Year 1.

The intention was for the team to develop technology and content using an agile software development process on a single server site at Purdue. It has been more difficult than expected to set up an iterative development environment and workflow. A major difficulty was in getting the necessary access for the team at the internal Purdue server site. Once we had managed our way through the bureaucracy, it was believed that there was insufficient time to use this capability so it was abandoned for this first year. Another challenge is that academic researchers are not necessarily full-time and it is very difficult to find frequent synchronization points. As a result we had the challenge of not having each of the development teaming working in the same environment with the same releases of code. We also had a rather adhoc version control system which consisted of uploads/downloads to/from DropBox. For the next year, we need to quickly establish a single site for integrated development work and a robust version control system.

As the architecture and design stabilizes, we will need to go through a 'configuration management' cycle such that all key members are notified of the change and have the opportunity to provide feedback including an assessment of how those changes might impact the work in their area going forward -- before the decision to adopt the change is made. To achieve proper communication, the change description needs to include high-level purpose and rationale; and key members of the team must be allowed to, and be expected to, provide an approval or disapproval with rationale. This needs to be an agile [quick] process, but without this configuration management and team input approach, suboptimal decisions might be made and poorly communicated. In recognition of this, the team will put together and follow a configuration management process for year one, option.

Another lesson learned is that the various pieces of the EA are closely linked and it is difficult to separate them. There has been some success in creating interfaces such that artifacts and dialog can be created without the involvement of developers. This has allowed the teams to work rather independently. However, there is still work that needs to be done to improve upon these interfaces and make them more robust and extensible.

In addition, we have not created the infrastructure which allows the content creators to test what they are developing as it is being developed. Also, we do not yet have all the project simulation work complete so we are not able to explore the results and create dialog to fit them, but rather are imagining what the simulation results might look like and creating dialog to support this. In the next year, we will need to create the capability to iteratively design content while experiencing the results. To support this, it will be beneficial to create the support to be able to stop, backup and restart an Experience to iteratively develop content based simulation results.

7.1.5 SPONSOR INVOLVEMENT:

The active involvement of the RT16 sponsors has been a major benefit for this program. In particular, the sponsors have provided expert feedback both in direction and with a larger number of resource and reference materials. Looking forward to the next year, there should be opportunities for the sponsors to become increasingly engaged as we further refine and evaluate the Experience Accelerator. Thank you for your support!

7.2 NEXT STEPS

The plan is to preserve most of the EA team going forward, with the addition of Dr. George Kamberov from Stevens, as shown in Figure 6. The team plans to leverage subject matter experts (SMEs) as applicable to the content to be developed in the next phase of the project.

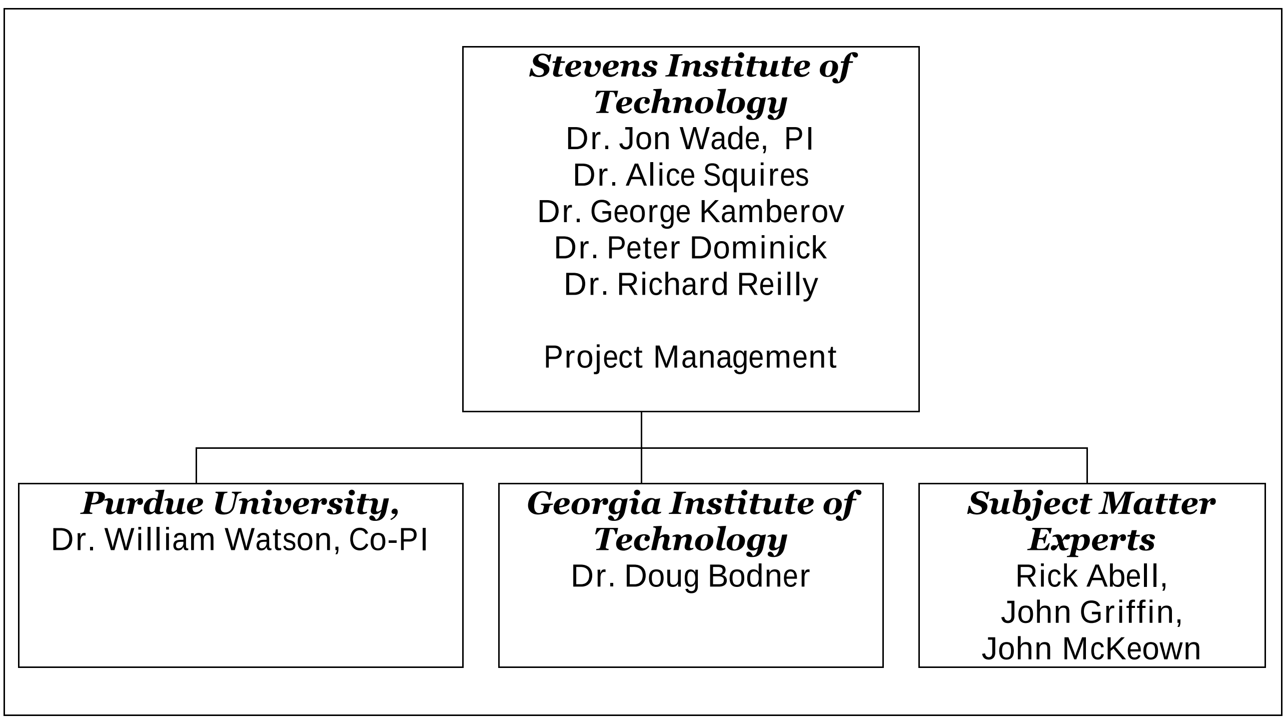

Figure 6: SERC RT16 Organizational Chart

Follow-on work has been defined for the next year in three phases:

- Phase 1: Update Documentation and Planning (June 2011)
- Phase 2: Prototype Development
- Phase 3: Prototype Validation

Phase 2 and 3 will overlap as needed during the months of July 2011 through May 2012. Work for Option 2 and beyond will involve measuring and analyzing pilot results. For more detail on the follow-on work plan, see *Developing Systems Engineering Experience Accelerator (SEEA) Prototype and Roadmap, Option Year 1 (Phase 1, Phase 2 and Phase 3) Technical and Management Work Plan.*

REFERENCES

Bagg, T. C., Granata, R. L., Brumfield, M. D., Casey, C. A., & Jamison, D. E. (2003). "Systems engineering education development (SEED) case study." In *Proceedings of the 13th annual international symposium, INCOSE 2003, Crystal City, Virginia, USA, June 29 - July 3, 2003.*

Charette, R. (2008). "What's wrong with weapons acquisitions?" *IEEE Spectrum*, November.

Defense Acquisition Program Support (DAPS) Methodology. (2009). *Defense Acquisition Program Support (DAPS) Methodology* (2nd ed. p. 442). Washington, DC: Office of the Deputy Under Secretary of Defense for Acquisition and Technology, Systems and Software Engineering. Retrieved from http://www.acq.osd.mil/se/docs/DAPS_V2.0_Methodology.pdf

Dubey, R. (2006) "Study and Analysis of Best Practices for the Development of Systems Engineers at a Multi-National Organization." *Thesis*, MIT, Boston, MA, USA.

Gavito, V., Verma, D., Dominick, P., Pennotti, M., Giffin, R., Barrese, T., et al. (2010, December 13). "Technical leadership development program: Final technical report SERC-2010-TR-013." Stevens Institute of Technology, Hoboken, NJ.

Kolb, D. (1984). *Experiential Learning: Experience as the Source of Learning and Development*. Prentice-Hall, Englewood Cliffs, NJ.

NDIA SE Division (2010, July). "Top systems engineering issues in department of defense and defense industry (Final 9a-7/15/10)."

Ryschkewitsch, M., Schaible, D., & Larson, W. (2009, January 20). "The art and science of systems engineering: Long version." *NASA*. Retrieved March 1, 2009, from http://www.nasa.gov/pdf/311198main_Art_and_Sci_of_SE_LONG_1_20_09.pdf

Squires, A. (2007). "Qualifying Exam: Systems Thinking and K12 Education." *Stevens Institute of Technology,* Hoboken, NJ.

Squires, A., Wade, J., Watson, B., Bodner, D., Okutsu, M., Ingold, D., Reilly, R., Dominick, P., Gelosh, D. (2011), "Investigating an Innovative Approach for Developing Systems Engineering Curriculum: The Systems Engineering Experience Accelerator", Proceedings of the 2011 American Society for Engineering Education (ASEE) Annual Conference and Exposition, Vancouver, BC, Canada, June 26-29, 2011.

Squires, A., Wade, J., Dominick, P., Gelosh, D. (2011) "Building a Competency Taxonomy to Guide Experience Acceleration of Lead Program Systems Engineers", Proceedings from the Ninth Annual Conference on Systems Engineering Research (CSER), Redondo Beach, CA, April 14-16, 2011.